

コンピュータ基礎演習

第2回

理工学部 情報科学科 隅田 康明

sumida@ip.kyusan-u.ac.jp

動画での説明内容

- 動画では、講義資料全ては説明しません
 - 主にレポートに関することを説明
 - **講義資料（PDF）をちゃんと読むこと**
- 講義資料だけみて、自分でレポートを出せるなら、動画は見なくても問題ありません
- 動画で説明する内容
 - 重要な注意事項
 - 動画があると分かりやすい説明
 - レポート内容についての説明
- 動きのあるものについては、一部字幕付きで別動画をアップロード

レポートの締め切りを変更

- 前回：講義日の6日後の23:59まで
- これから：講義のあった週の**金曜日の18:00**まで
 - 補講等で月曜日以外の講義の場合は別途指示
- 変更理由：日曜日夜中に提出する学生がいたため(数名ですが)
 - 多少のトラブルがあっても余裕を持って取り組めるように、との配慮で締め切りを長くしていたが、結果として日曜日の夜中に作業する学生が出たため
 - 元々、土日に作業をさせるつもりはない
 - 無理をさせないための長めの締め切りなのに、夜中に作業するなら本末転倒なので

この授業の履修について

- 遠隔授業の延長に伴って、
PC無しでもレポートを作成できる内容に変更します
 - シラバス変更に不満があれば、履修を取り消しましょう
- ただし、スマートフォンも無い場合は単位取得が不可能なので、
履修を取り消しておきましょう
 - 来年、履修してください
 - 今年取り消して、来年人数制限などで履修出来ない場合は、私から教務にかけあいます

Processing等のインストール

- いくつかアプリをインストールする必要があります
 - PCの場合：最低10GB程度の空き容量を確保
 - スマートフォンの場合：最低1GB程度の空き容量を確保
- 今回の演習で、Processingを動作させられなかった場合
 - アプリケーションインストールに失敗 かつ
OpenProcessingでのサンプル実行に失敗
 - そうそうないと思いますが
 - Processingを動作させられないのであれば、
レポートを出せない → 単位を取れない
 - 今年の履修は取り消して、来年受講しましょう

インストール予定のアプリ

- (進捗によっては使わない場合もあり)
- Outlook : スマホでメール管理
- PowerPoint : レポート作成に使用
 - 大学アカウントでサインインを前提
- Processing : OS毎にそれぞれインストール
- Word : Wordでレポート提出で加点する回を予定
- Excel : 座標計算や色計算の補助として使用を検討中
- その他、画像加工アプリ : スマートフォンで作業の場合
 - 課題作成時に、フリーの画像加工アプリでProcessingで作ったCGを加工し、説明することで工夫点の一つとする

講義用HPについて

- 講義資料や、講義に必要な情報は、この授業用のホームページに掲載しています
 - K'sLifeの授業共有ファイルにもアップロードしていますが、HPの方がアクセスはしやすいはずです
 - K'sLifeに接続しにくい状況が続いています
 - 念の為、K'sLifeにもアップロードはしますが、基本的には講義HPを見るようにしましょう
- 下記のHPから、講義資料、動画のURL、Zoomの招待リンクなどを確認できます。

<http://www.is.kyusan-u.ac.jp/~sumida/class/pckiso/>

遠隔期間中の出席について

- 出席：期限内のレポートの提出で出席とする
 - 出席とレポートの点は別なので注意すること
 - 手抜きレポートでも出席にはなるが、レポート点を取れなければ評点が足りなくて単位を取れなくなる
 - 出席するだけ、では単位は取れない
 - 期限を過ぎたら遅刻で登録
- 配慮すべき事情があって、事前に間に合わない報告・相談があった場合には、締切後も受け付ける（出席になる）
 - 当たり前だが、レポートを出さなければ欠席
 - 特に理由もなく、レポートに取り組むのが遅れた場合等は除く
 - 早く相談しなければ欠席(遅刻)または減点

レポートの提出と締め切り

- 提出方法：毎回指示する（毎回同じではない）
 - 第1回と2回(今回)：メールで提出
 - メールでの質問方法を身に付けさせる目的
 - 第3回以降は別の提出方法に変更
 - 見落としを減らすため、メールは質問受付のみにしたい
- 締め切り：原則、講義のあった週の**金曜日の18:00**まで
 - 補講日などで月曜以外が講義日の場合は別途指示する
 - ただし、配慮すべき事情があれば遅れても減点はしない
 - 配慮すべき事情については次頁を参照
 - 遅れ提出：最終講義日までは遅れ提出も受け付ける
 - ただし、3回以上で最高評価をBに制限
 - 真面目に期限内に提出した学生との差別化。そもそも、遅れ提出が多いとレポートの減点も多くなるので、C以上の評価を取ることが難しくなる場合がほとんど。

出席とレポートについての注意

• 出席とレポートの点数は別

- 出席になったからといって、レポートの点も満点とは限らない
 - 出席自体は単位取得要件を満たすだけ、**単位を取れるかはレポートで決まる**
 - **手抜きレポートは当然減点する**
 - レポートの点が足りなければ不可になるので、ただ出せばいい、とは思わないこと

• レポートが複数ある場合もあるので気を付けること

- 今回はメールだけだが、次回は複数出題

遠隔授業期間中の質問

• 困ったら早めに質問・相談！！

- 学生側から質問されないと、誰が困っているのか、何が分からないのか、分かりません
- メールで質問：やり取りに時間はかかるが一番確実
 - メールで質問→Zoomで指導を提案する場合もある
- Zoomで質問：授業時間中限定
 - 時間は限られるが、作業中の画面を見ながら教えられるので、問題を短時間で解決出来る可能性が高い
 - 簡単に解決できない場合もあるので、その場合はメールのやり取りに移行するばあいもある
- Line OpenChatで質問
 - 文字だけのやり取りに限定（画像アップロードは禁止）
 - 質問内容が他の学生にも分かるので注意すること

授業についての質問メールの書き方

[授業名(曜日時限)]についての質問	} 件名
～先生	} 誰宛か
[授業名(曜日時限)]を受講しています、 20AA999の九産太郎です。	} 何者か
(質問内容)	} 用件
--	
20AA999 九産太郎 九州産業大学 芸術学部 ○○学科 1年	} 署名

相手に丁寧に対応して欲しいなら、自分も丁寧に！

Zoom、OpenChat

- アクセスは講義HPから：今回は通知も出しますが
 - <http://www.is.kyusan-u.ac.jp/~sumida/class/pckiso/>
- どちらも、名前を「学籍番号 氏名」とすること
 - 変更していない場合は、質問には答えない
 - 名前を変えてから質問しましょう
- Zoom、OpenChatで質問するかは自由
 - 質問に来ても出席にはならない
- どちらも対応は講義時間中に限定
 - 時間外は不可、と言うか繋がらない
 - ただし、OpenChatで時間内に処理出来ない質問があった場合は、後でメールで対応する

Office365にサインインできない

- 既に別のアカウントでサインインしている
 - 送信がOutLookになってる場合は大抵このケース
 - 使っていないアカウントなら、サインアウトしておく



大学メール専用ブラウザを作ってしまう

- (複数アカウントがある人はどちらかと言うとこっちを推奨)

Office365サイトにアクセスしたら、
OutLookアカウントでサインインしているケース
以降は例。自分の環境に合わせて設定すること。

- Edge  : Outlookメールアカウント
- Chrome  : 大学アカウント用 などで使い分け
 - https://www.google.com/intl/ja_jp/chrome/
- その他、よく使われるブラウザ
 - FireFox : <https://www.mozilla.org/ja/firefox/new/>

スマホにOutLookアプリをインストール

- (これは次回のレポートの1つ)
- 普段はインストールを“強く推奨”程度ですが、
事態が事態なので、次回の演習の1つにしてしまいます
 - 大学からのメールをスマホで使えるとすごく便利なので
- インストール方法
 - http://www.is.kyusan-u.ac.jp/~sumida/lecture/OutLook_install.pdf
- 早くインストールしておいた方が役には立つので、
時間がある人は次回の先取りでインストールしておきましょう

PCの購入について

- いくつか質問があったので全体向けに簡単に
- この授業のためにPCを購入する必要はありません
- 購入するなら、今後の大学生活を考えて購入しましょう
 - **まずは芸学の先生（クラス担任など）に相談**
 - 芸術学部推奨PCも販売されています
 - 選択肢の一つとして提示するだけで、私が推奨している訳ではないですが
 - https://www.kyusan-u.ac.jp/nyushi/news/news_1388

今日の授業内容

- シラバス変更の説明
 - 講義概要・到達目標、評価
- **Processingのインストール**
 - Windows, Mac, iPhone, Android
それぞれの環境に合わせたアプリをインストール
- **スクリーンショットの撮影と画像のメール添付**
 - Processing実行画面のスクリーンショットをメールで送信
- Processingについては、また後日(まずは準備を優先)

今回のレポート

- メールでProcessingの実行画面を送信
 1. Processingのインストール
 - アプリインストール権限の問題でインストールできない場合などは、遅れても待つのでまずはメールで相談すること
 2. スクリーンショットの撮影
 3. メールに画像添付
 - 目的：今後のレポート作成準備

シラバス変更点の説明

- 遠隔授業期間延長に伴い、シラバスを変更
 - PC未所持学生でも実施可能な内容に変更
 - PC、スマートフォン用の資料を提示、各自の環境に合わせてレポートを作成する
 - 講義準備回を数回導入
 - 必要なアプリケーションのインストールなど
 - スマートフォンで長いプログラミングを行うことが困難になるため、雛形の変更を主にした内容に変更
 - コピペ → ちよつと修正 → レポートで工夫点を説明
 - PC所持かつ、より高度な内容を希望する学生には、別に資料を配布する
 - 物足りない学生向けの自学自習用の資料
 - 提出すれば加点はするが、成績に直結するものではない

更に再延長された場合、再変更はしない

講義概要

- 本演習ではProcessingによるデジタルアート制作を通して、コンピュータまたはスマートフォンの基礎的な使用方法を学ぶ。演習では、Processingによるプログラミングの基本を学習し、プログラミングによるコンピュータグラフィックス制作を実践する。また、制作した作品についてのレポート作成を通して、ビジネス系ツールであるプレゼンテーションツールの基本的な操作方法を学習し、ビジネスに必要とされる一般的な情報リテラシー、およびネットワークリテラシーを理解する。本授業科目は、芸術学部のDP2及びCP2に基づいて必要な技術・技能を習得することを目的として設置している。本授業科目は、芸術学部共通の基盤を構成するIT力育成科目として、1年次に配当されている。

講義計画 (1~7)

回	内容
1	ガイダンス (遠隔授業)
	受講上の注意、スマートフォンでのメール送信
2	遠隔授業の受講準備：Processingのインストール
	Processingのインストール、スクリーンショットの撮影、メールに画像を添付
3	Officeアプリケーションのインストールと実行 (レポート作成準備 (1))
	OutLook、PowerPointインストール、簡単なレポート作成
4	図形の組み合わせによるCG制作
	Processingで静止画を作成：座標指定による図形配置、スクリーンショットの提出
5	アニメーションと入力イベント処理
	動く図形の描画、マウスやキーボード操作による図形描画
6	ランダムとノイズを使ったジェネラティブアート制作入門
	条件に応じた図形の移動や変化
7	条件分岐 (1) : if文、if-else文の使い方
	条件分岐を利用したプログラム、条件によって変化する絵

講義計画（8～13＋追加課題）

回	内容
8	条件分岐（2）：当たり判定、簡単なゲームプログラミング
	簡単なゲーム作成
9	繰り返し（1）座標変換：for文の使い方、座標変換の考え方
	繰り返しと座標変換による図形の描画
10	繰り返し（2）：複数の図形を動かす
	繰り返しによる複数の図形の描画とアニメーション
11	デジタルアート制作（1）
	オリジナルの作品制作：構想～制作活動
12	デジタルアート制作（2）
	オリジナルの作品制作：制作活動
13	デジタルアート制作（3）
	オリジナルの作品制作：レポート作成
追加課題	追加課題の出題：授業回数減に伴う措置
	最終的な評点に最大5点加点、ただし、評点は100点満点

40点

シラバス変更点

- 講義概要
 - ワードプロセッサの基本的操作方法を削除
 - スマートフォンでの操作が困難
 - PowerPointなら最低限の作業は可能と判断
 - ただし、レポートをWordで作成してもよし
 - 文章を一部修正：本質に関わる習性は無し
- 講義計画
 - 高度な内容を削除し、初回～3回までを講義準備回に変更

評価方法（前回説明から変更）

- 演習点100点で評価
 - 通常課題：50点（1～12回）
 - ~~コード集、作品集：10点~~ スマホでの作業が難しいため廃止
 - ~~通常課題1回につき1点(加点もあり)~~
 - **ミニテスト：10点** 代わりに実施
 - 解き直し可のミニテストをコード集の代わりに追加
 - 点数に応じて0～10点で評価する
 - 制作課題：40点（11～13回）
 - 第13回提出のレポートで0～40点になるが、11回・12回のプログラムも通常課題点が付くため、提出しなければ不可となる
- **出席回数が著しく少ない場合：E評価(単位なし)**

使用する端末について

- アップデートは最新にしておく
 - Windows7・8の場合は仕方がないが、既にサポート切れのOSであり、何かあっても自己責任であることは理解しておくこと
 - 買い換えろ、とまでは言いませんが、他と比べて危険性が高いOSであることは理解しておきましょう
- ブラウザはChromeの使用を前提に解説
 - PC、iPhone、Androidいずれも
 - 入れていない人はインストールしておきましょう
 - 別のブラウザでも良いが、その場合は自己責任で

演習（レポートの手順）

- ここからは、各自の端末に合わせて資料を見て作業すること
 1. Processingをインストール
 2. プログラムを入力して実行（一部最初から入力済み）
 3. 実行結果のスクリーンショットを撮影
 4. メールにスクリーンショットを添付して送信

今回入力するプログラム

- 端末によって初期入力されていたりする
 - 基本はOSごとの説明資料に書いてある通り入力
 - 実行できればアレンジしても良い
 - 消えてしまったらこれを入力する

```
void setup(){
  size(400,400);
}
void draw(){
  ellipse(mouseX, mouseY, 10, 10);
}
```

OSごとの資料にも載せているが、読みにくい場合にはこちらを見ること

プログラムの説明(少しだけ)

動くプログラムを作るときに必ず書く定型文

```
void setup(){  
}  
void draw(){  
}
```

```
ellipse(mouseX, mouseY, 10, 10);
```

- ellipse(x, y, 幅, 高さ); は楕円を描く命令
- (mouseX、mouseY) の位置に幅10高さ10の円を描く
 - つまり、マウスがある位置に円を描いていく

少しプログラムを書いてみたい人向け

- これは次回以降の内容の先取り: 余裕があれば試してみよう

```
void setup(){
  size(400,400);
  background(0,0,255);
}
void draw(){
  ellipse(mouseX, mouseY, 10, 10);
}
```

- `background(0,0,255);` //背景の色を青色にする
 - ()の中の3つの数字は、(赤,緑,青) の色の強さ
 - それぞれ0~255で指定する

アレンジ例 (2)

```
void setup(){
  size(400,400);
}
void draw(){
  fill(255,0,0,20); //塗りつぶしの色を半透明の赤にする
  ellipse(mouseX, mouseY, 10, 10);
}
```

- fill(赤,緑,青,透明度); で塗りつぶしの色を変えられる
 - それぞれ0~255で指定する
- // はコメント：これは書かなくて良い
 - 書いてもいいが、全角モードにならないように気を付ける

アレンジ例 (3)

```
void setup(){
  size(400,400);
}
void draw(){
  stroke(0,255,0,50); //線の色を緑にする
  fill(255,0,0,20); //塗りつぶしの色を半透明の赤にする
  ellipse(mouseX, mouseY, random(20), 10);
}
```

- stroke(赤,緑,青,透明度); で線の色を変えられる
 - それぞれ0~255で指定する
- random(20) は, 0~20のランダムな数を得る命令
 - この場合だと、円の横幅が毎回ランダムに変わる

プログラム入力時の注意

- 入力するプログラムは、**全て半角英数**
- **大文字小文字**、**カッコの形**、**記号の形**に気を付ける
 - 大文字、小文字は別の文字として区別される
 - **一文字違うだけでも動作しない**こともあるので、打ち間違えないように気を付けること
- 間違えやすい記号
 - **[,]**と**[.]** 今回は、カンマ [,]のみ。[.]は使わない
 - **[;]**と**[:]** 今回は、セミコロン[;]のみ

日本語入力(ひらがなモード) は使わない！！

スクリーンショット

- スクリーンショット
 - 画面の全体または一部を写した画像のこと
 - スクショ、スクリーンキャプチャ、画面キャプチャ等の呼び方もされる
 - 見ている画面をそのまま画像にすることが出来る
 - カメラ撮影と違い、歪みもなく綺麗に撮影できる
- レポートの提出だけでなく、「こんな画面が出てきたんですが」、「こんな表示が出てるんですが」の質問に対応するときに、画面のスクショがあると状況の把握が簡単になる
- 必ずスクショ撮影→メール送信を出来るようになっておくこと

スクリーンショットの撮影方法

- 資料では代表的な操作例の1つを紹介しているが、他にも色々、スクショ撮影のやり方はある
 - 例えば、Windows編では、7や8でも実行できる方法を載せているが、Windows10なら
[Windows] + [Shift] + [S] が一番簡単
- 他のやり方は自分で調べてみよう
 - 特にAndroidは機種ごとに色々違いがあるので、上手くいかない場合は、ある程度自分で調べるようにしましょう
 - 「機種名 スクショ 撮影」などで検索

レポートチェックリスト（第2回）

- Processingをインストールした
- プログラムを入力して実行した
- 実行結果のスクリーンショットを撮影した
- メールにスクリーンショットを添付して送信した
 - 差出人：k学籍番号@st.kyusan-u.ac.jp
 - 宛先：sumida@ip.kyusan-u.ac.jp
 - 件名：PC基礎(月○)02
 - 本文：学籍番号と氏名、感想・質問
 - 添付ファイル：スクリーンショット画像

レポート 1

- 講義HPをブックマークした
- Formsのアンケートに回答した

レポート 2

今日のレポートの例

宛先 : sumida@ip.kyusan-u.ac.jp

件名 : PC基礎(月○) 02

20AA999の九産太郎です。
第2回の**レポートを提出**します。

Processing : Windows/Mac/iPhone/Android
その他、質問感想 : 何かあれば、自由に書いてください
(特になければなしでも構わない)

--

20AA999 九産太郎
九州産業大学 芸術学部 ○○学科 1年

先に掲載しておくが、出すのはスクリーンショットを撮ってから

ショートカット

- Processingインストール、実行、
スクリーンショット撮影、撮影画像のメール添付
 - [Windows編](#)
 - [Mac編](#)
 - [iPhone編](#)
 - [Android編](#)
- PDFの表示環境によっては、ジャンプ出来ない場合がある
 - その場合は、スクロールして自分の環境を探すこと

動画へのリンク

- 今回はPDF内に動画へのリンクを貼るが、次回からは講義HPかMoodle(次回説明)からにする予定
 - <http://www.is.kyusan-u.ac.jp/~sumida/class/pckiso/>
 - ↑を見れば全部揃っています
- 第2回の講義動画
- レポート解説
 - Windows版
 - Mac版
 - iPhone版
 - Android版

レポート

Processingのインストールと実行
スクリーンショットの撮影

Windows編

Processingをダウンロード



- Processingサイトにアクセス
 - <https://processing.org/>
- 左のメニューから「Download」
- 自分のPCのOSをクリック

Download Processing. Processing is available for Linux, Mac OS X, and Windows. Select your choice to download the software below.



3.5.4 (17 January 2020)

[Windows 64-bit](#)

[Linux 64-bit](#)

[Mac OS X](#)

[Windows 32-bit](#)

ダウンロードしたファイルの場所に移動

- ダウンロード中、ファイルサイズが大きいので少し時間がかかる

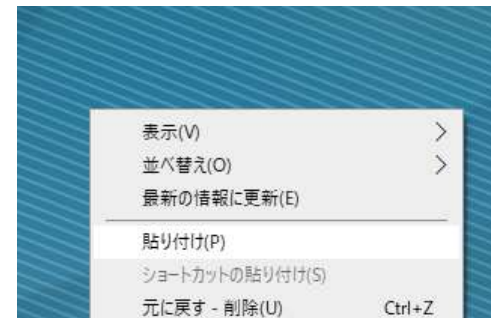
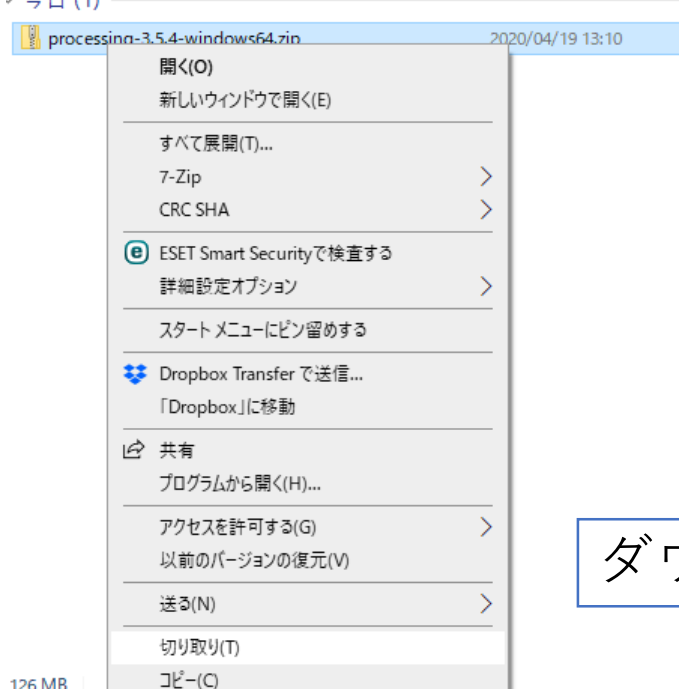


ダウンロード完了したら、「フォルダを開く」などで、ファイルのある場所に移動。

- ※寄付を求めるページが開くが、寄付は必須ではない
 - 寄付しなくてもProcessingは利用できる（ありがたいですね）

デスクトップにファイルを移動

- 右クリック → 切り取り → デスクトップに貼り付け



ダウンロードが完了してから移動すること！

ファイルを展開する

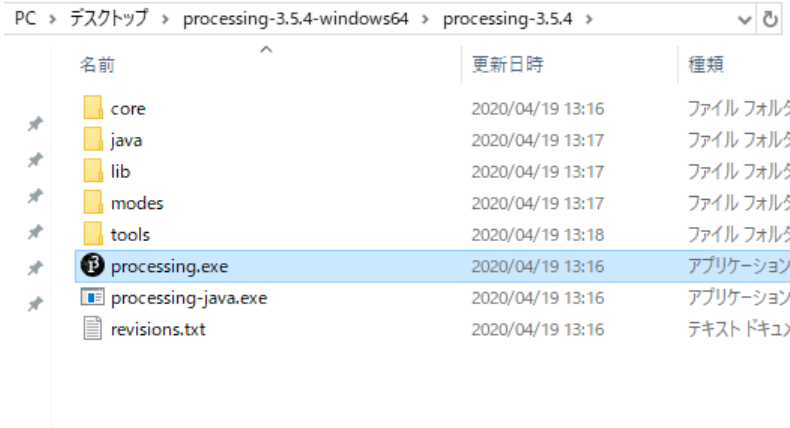
- 右クリック → すべて展開 → 展開



終わるまで待つ

Processingを起動

- 展開したら、[processing-3.5.4-windowsxx]
→ 「processing-3.5.4」と開き、
- 「Processing.exe」をダブルクリックして実行

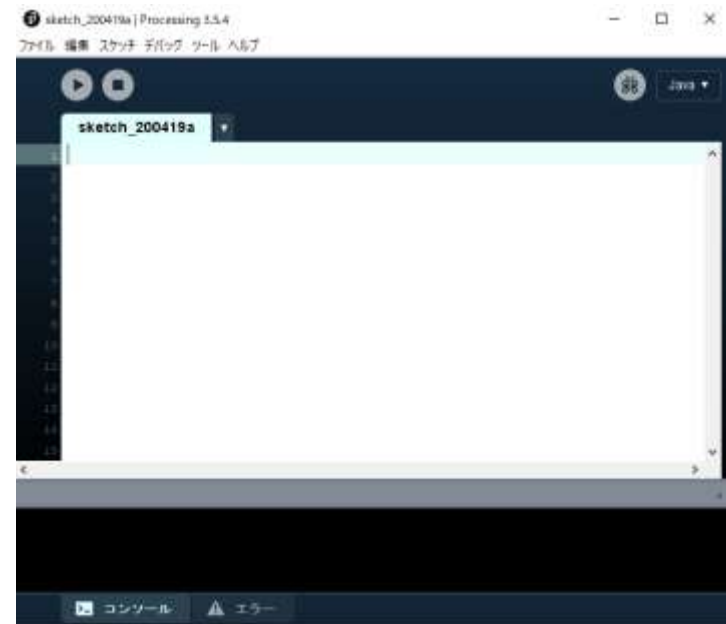
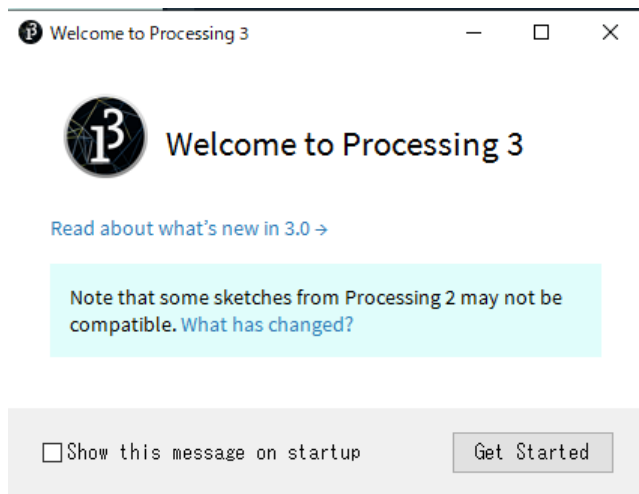


- ↓こんなのが出てきたら、「実行」をクリック



Processingの起動を確認

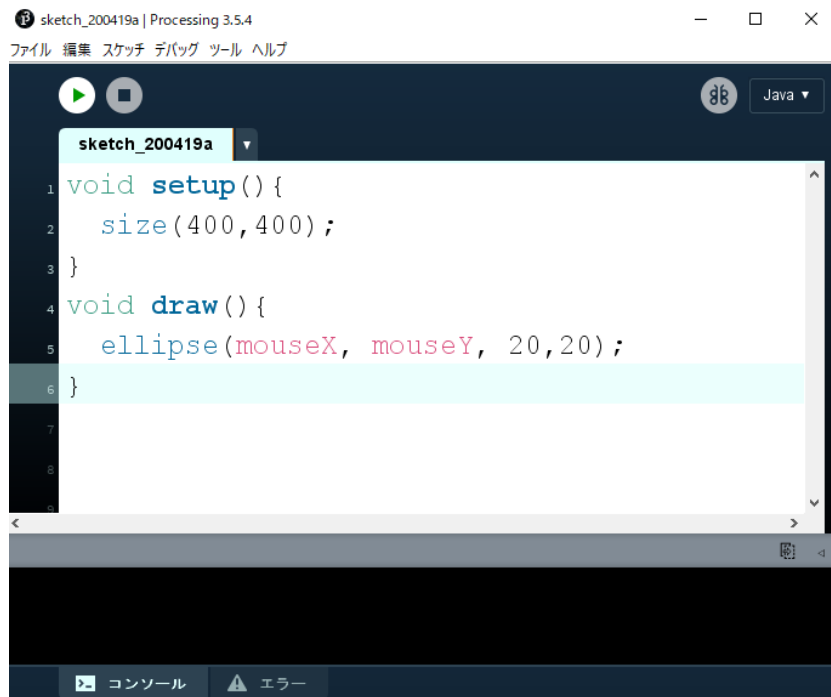
- ↓が出てきたら、「Get Started」



この画面が表示出来たら起動成功

プログラムを入力して実行

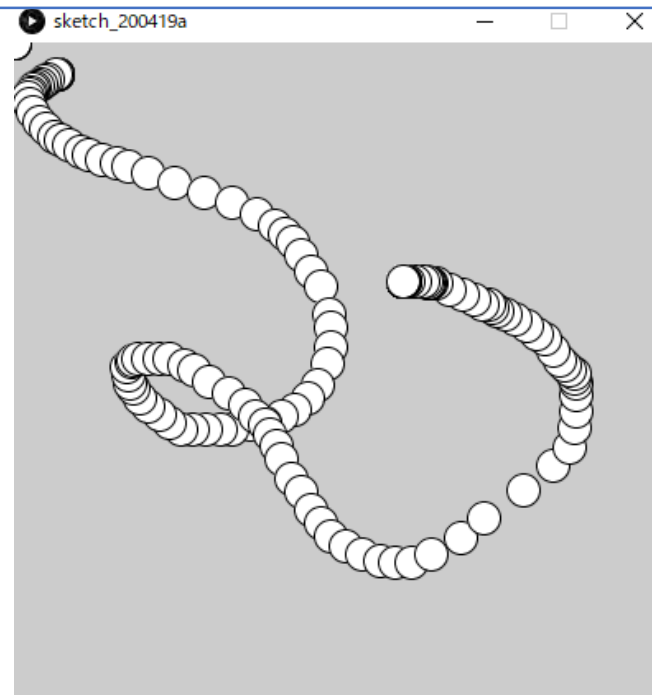
- この通りにプログラムを入力



```
sketch_200419a | Processing 3.5.4
ファイル 編集 スケッチ デバッグ ツール ヘルプ

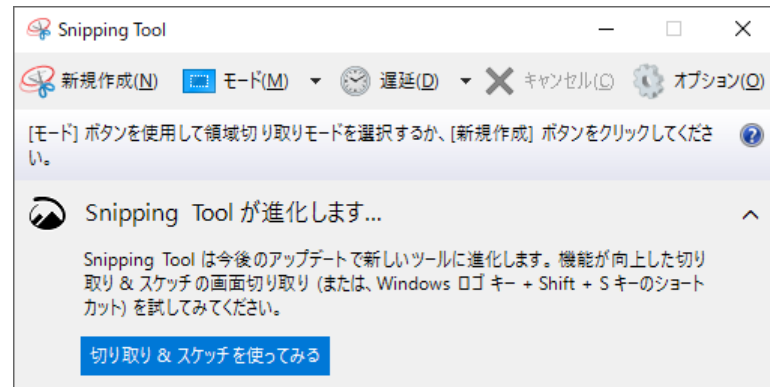
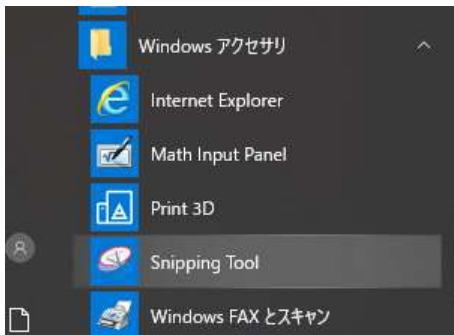
sketch_200419a
1 void setup() {
2   size(400,400);
3 }
4 void draw() {
5   ellipse(mouseX, mouseY, 20,20);
6 }
7
8
9
10
コンソール エラー
```

マウスで絵を描けることを確認



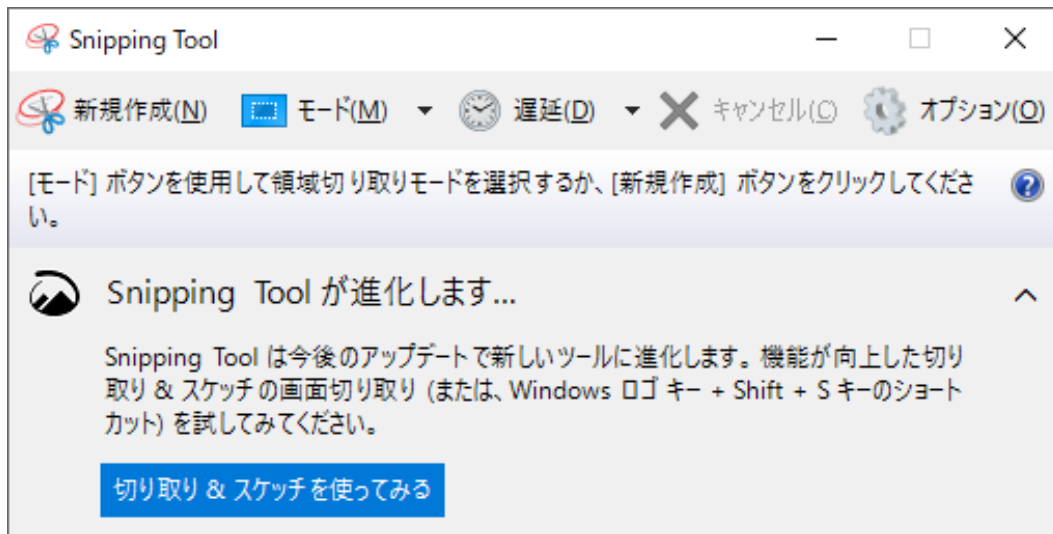
実行結果のスクリーンショットを撮影(1)

- Windowsの場合 : Snipping Toolを利用
 - スタート→「Windowsアクセサリ」→「Snipping Tool」をクリック



実行結果のスクリーンショットを撮影(2)

- 「新規作成」ボタンをクリック
 - 画面が白っぽくなるので、
 - 撮影したい領域をドラッグして選択

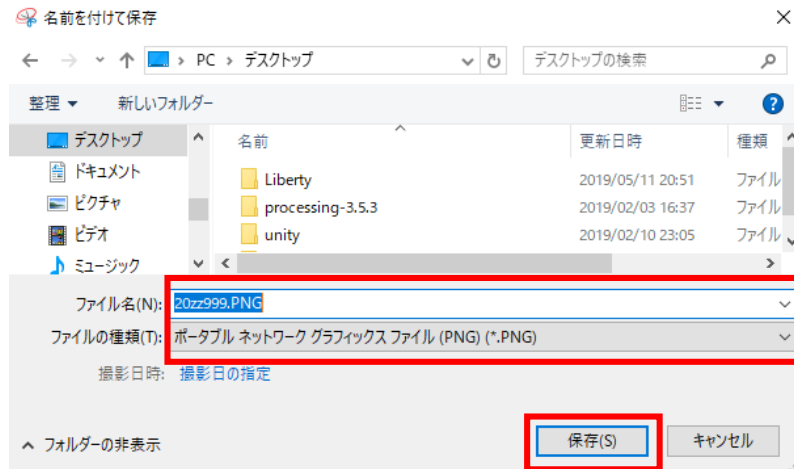


実行結果のスクリーンショットを撮影(3)

- 撮影した画像を保存

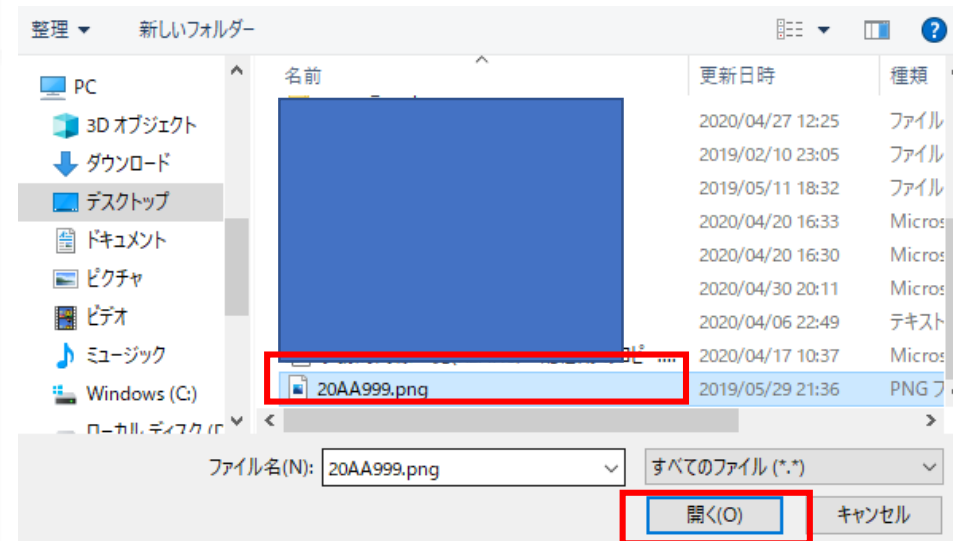
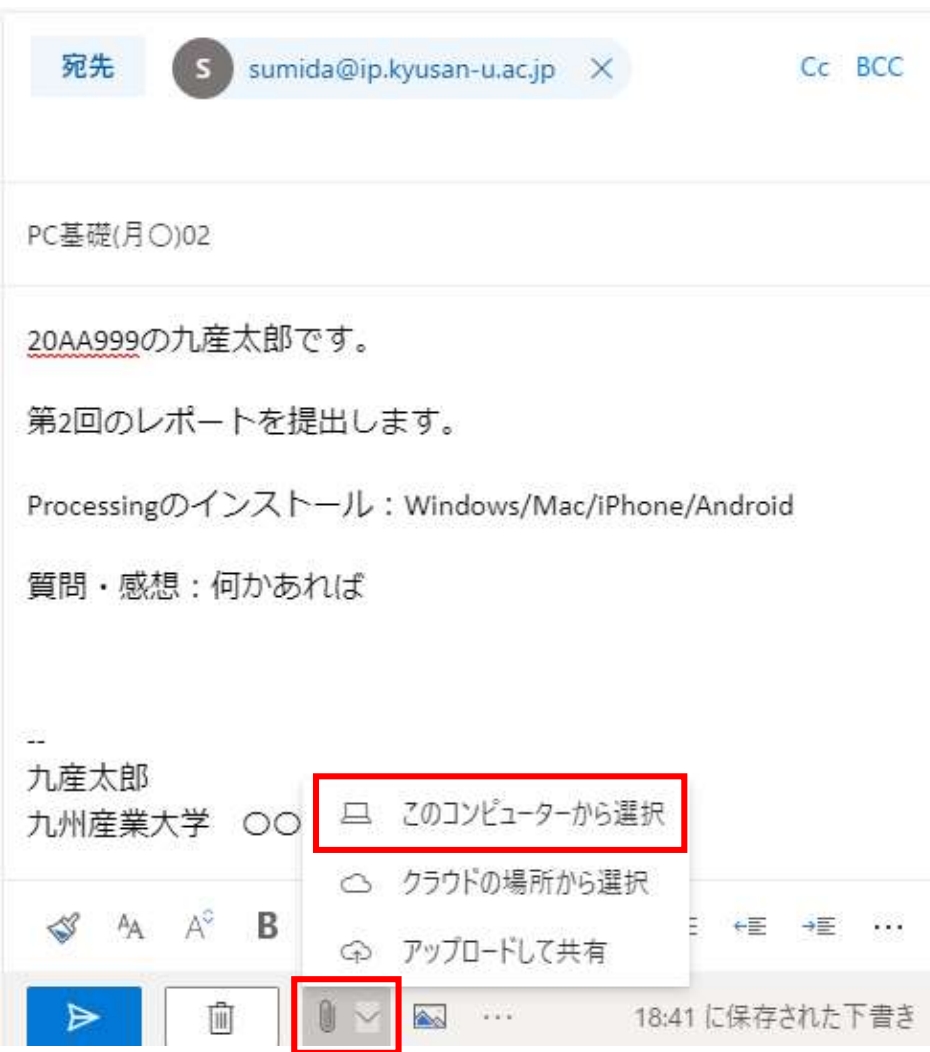


- ファイル名を学籍番号に変更して、「保存」



スクリーンショットをメールに添付

- メール作成画面の開き方は前回の資料を参照



保存したファイルを選んで「開く」

宛先

S

sumida@ip.kyusan-u.ac.jp X

Cc BCC

PC基礎(月〇)02



20AA999の九産太郎です。

第2回のレポートを提出します。

Processingのインストール：Windows/Mac/iPhone/Android

質問・感想：何かあれば

--
九産太郎

九州産業大学 〇〇学部〇〇学科〇年

📧 AA A° B 添付 U ✎ A ☰ ☷ +☰ +☷ ...



18:41 に保存された下書き

実行画面が写っているのを確認して、▶で送信

送ったら、自動返信メールが来るのを確認しておこう。

(返ってこない場合もあるので、送れている自信があればそれでも良い)

レポート

Processingのインストールと実行
スクリーンショットの撮影

Mac編

Processingをダウンロード



- Processingサイトにアクセス
 - <https://processing.org/>
- 左のメニューから「Download」
- 自分のPCのOSに合わせてダウンロード

Download Processing. Processing is available for Linux, Mac OS X, and Windows. Select your choice to download the software below.



3.5.4 (17 January 2020)

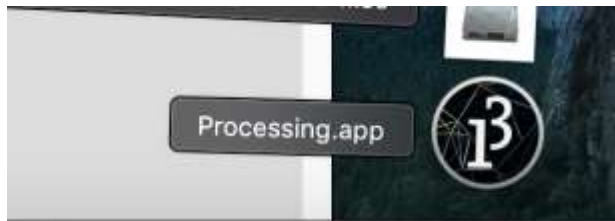
Windows 64-bit
Windows 32-bit

Linux 64-bit

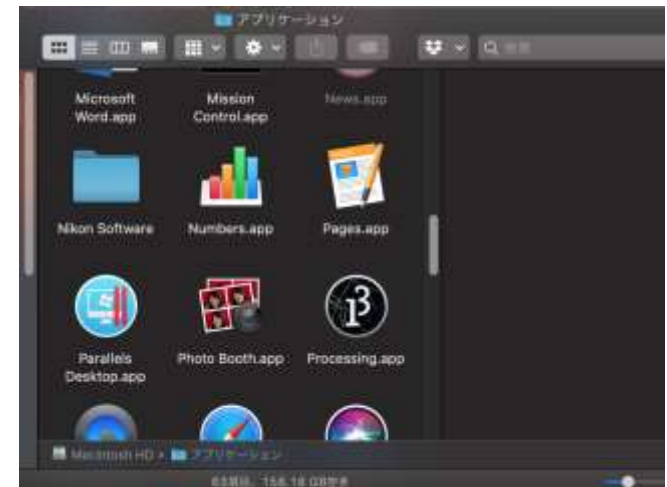
Mac OS X

ダウンロードしたファイルを解凍

- ダウンロードしたファイルをクリックして解凍



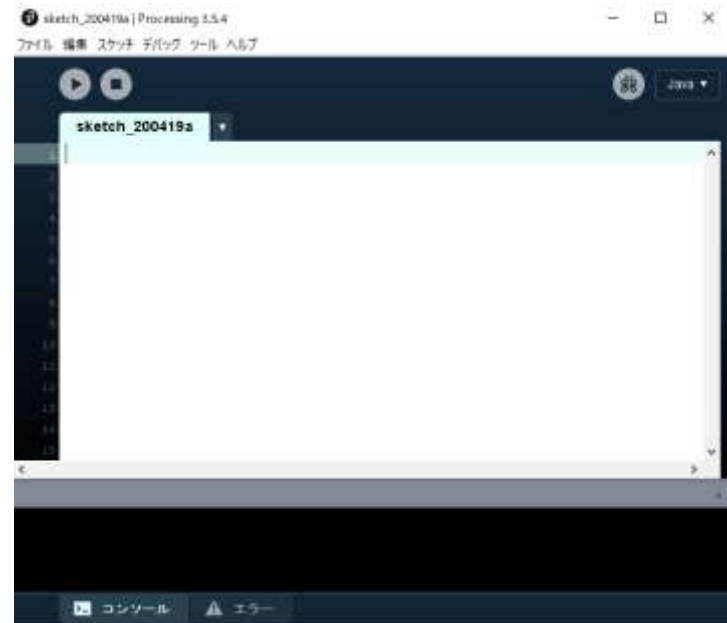
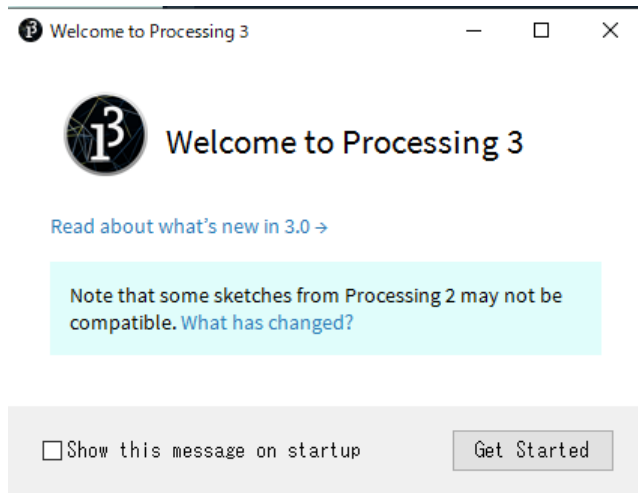
- 解凍したら
アプリケーションフォルダにコピー



Processingを起動

- アプリケーションフォルダの「Processing」をクリックして実行

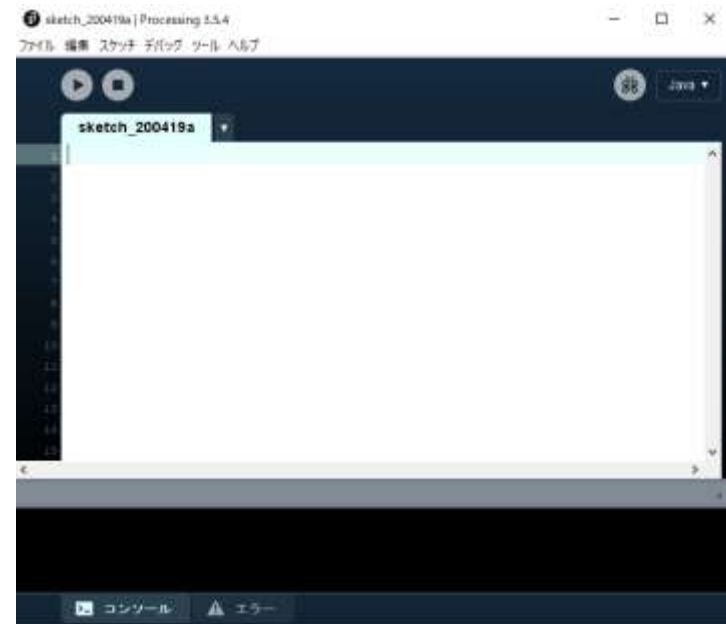
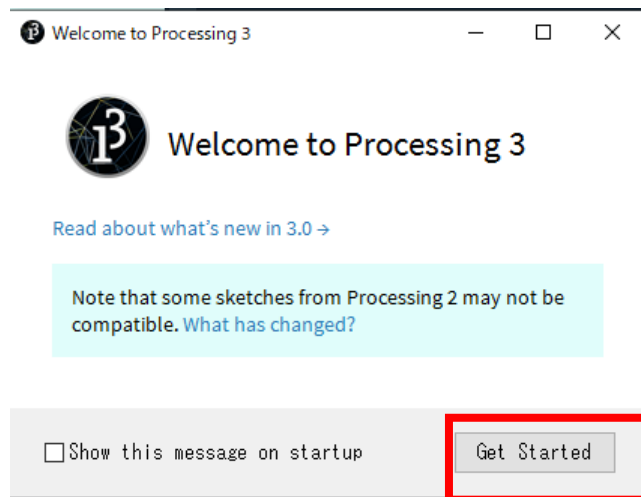
↓が出てきたら、「Get Started」



この画面が表示出来たら起動成功

Processingの起動を確認

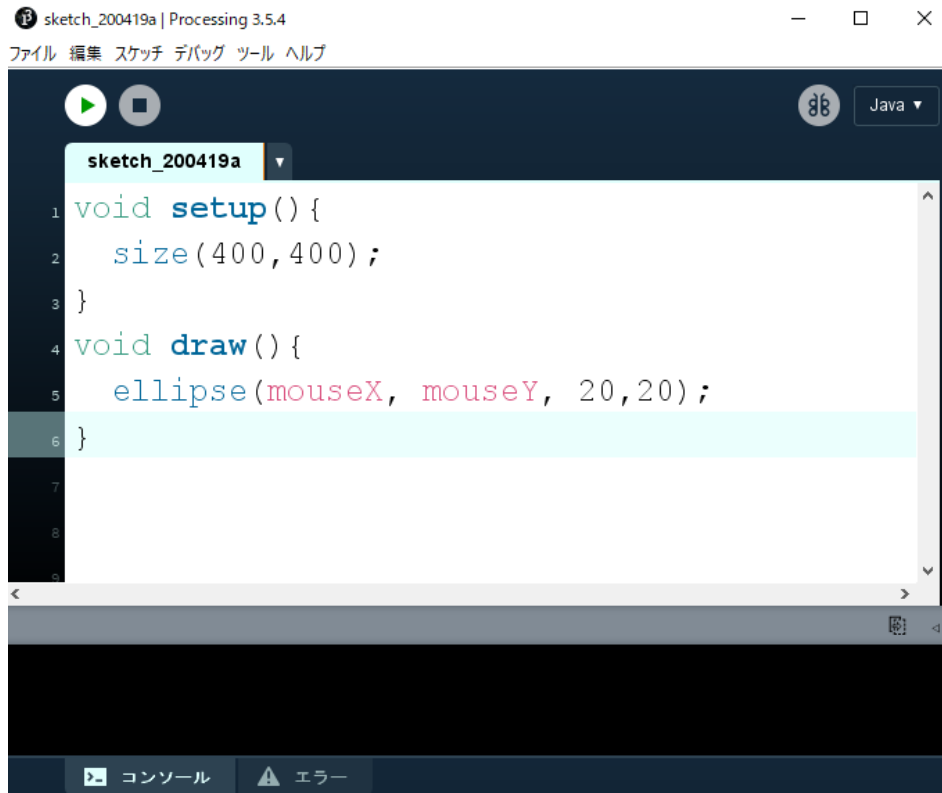
- ↓が出てきたら、「Get Started」



この画面が表示出来たら起動成功

プログラムを入力して実行

- この通りにプログラムを入力

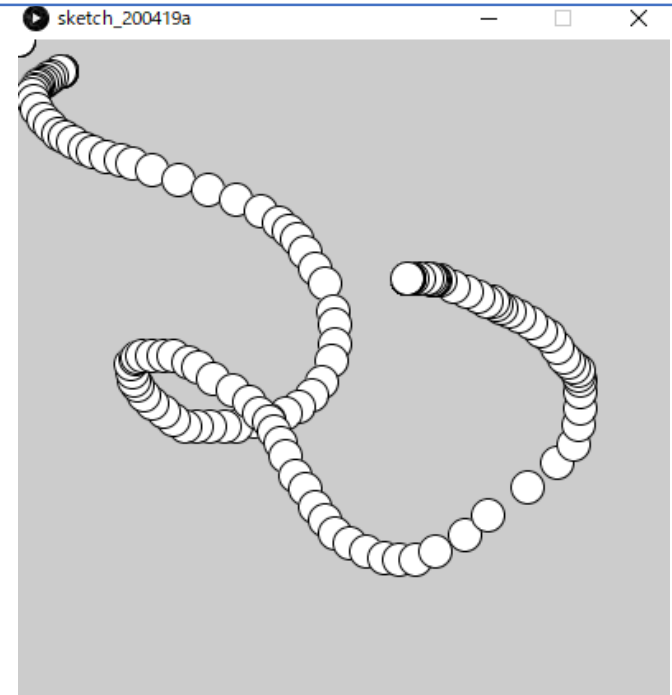


The screenshot shows the Processing IDE interface. The title bar reads "sketch_200419a | Processing 3.5.4". The menu bar includes "ファイル", "編集", "スケッチ", "デバッグ", "ツール", and "ヘルプ". The code editor contains the following Java code:

```
1 void setup() {  
2   size(400, 400);  
3 }  
4 void draw() {  
5   ellipse(mouseX, mouseY, 20, 20);  
6 }  
7  
8  
9
```

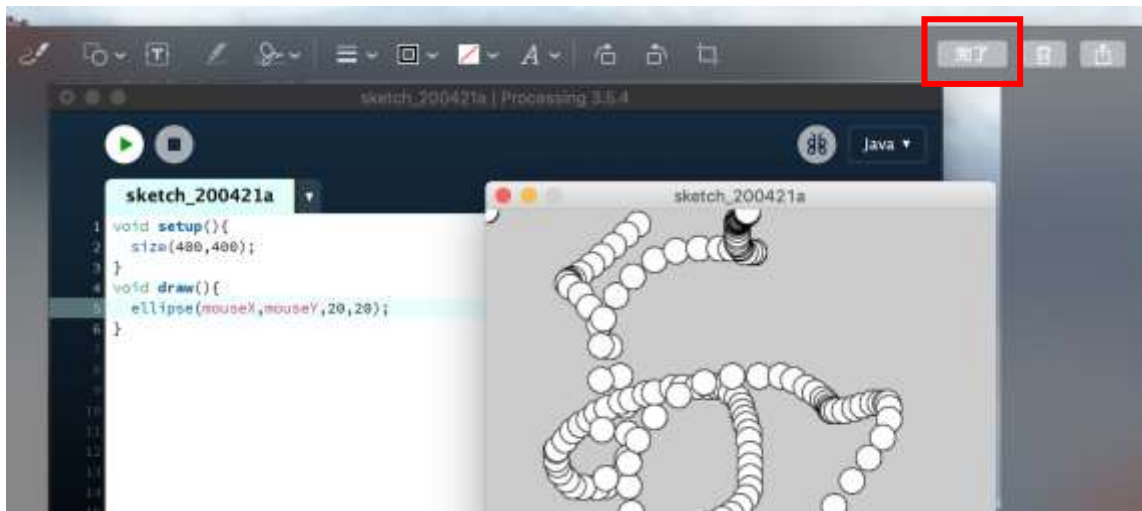
The code is highlighted in a light blue color. The IDE also shows a "コンソール" (Console) and "エラー" (Error) panel at the bottom.

マウスで絵を描けることを確認



実行結果のスクリーンショットを撮影(1)

- Macの場合： [command] + [shift] + [4] で撮影



- デフォルトなら、デスクトップに画像が保存される
 - 変更しているなら、変更した場所に画像がある

画像の名前を変更

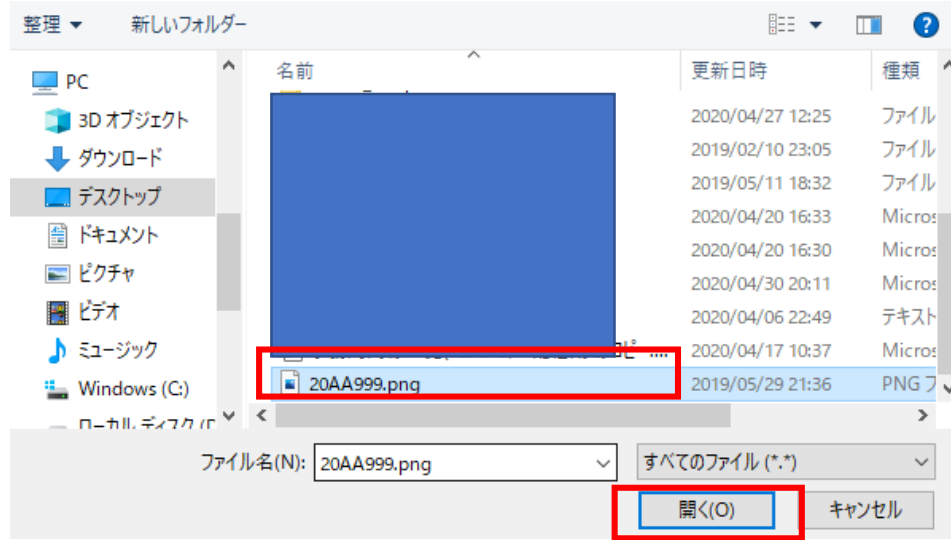
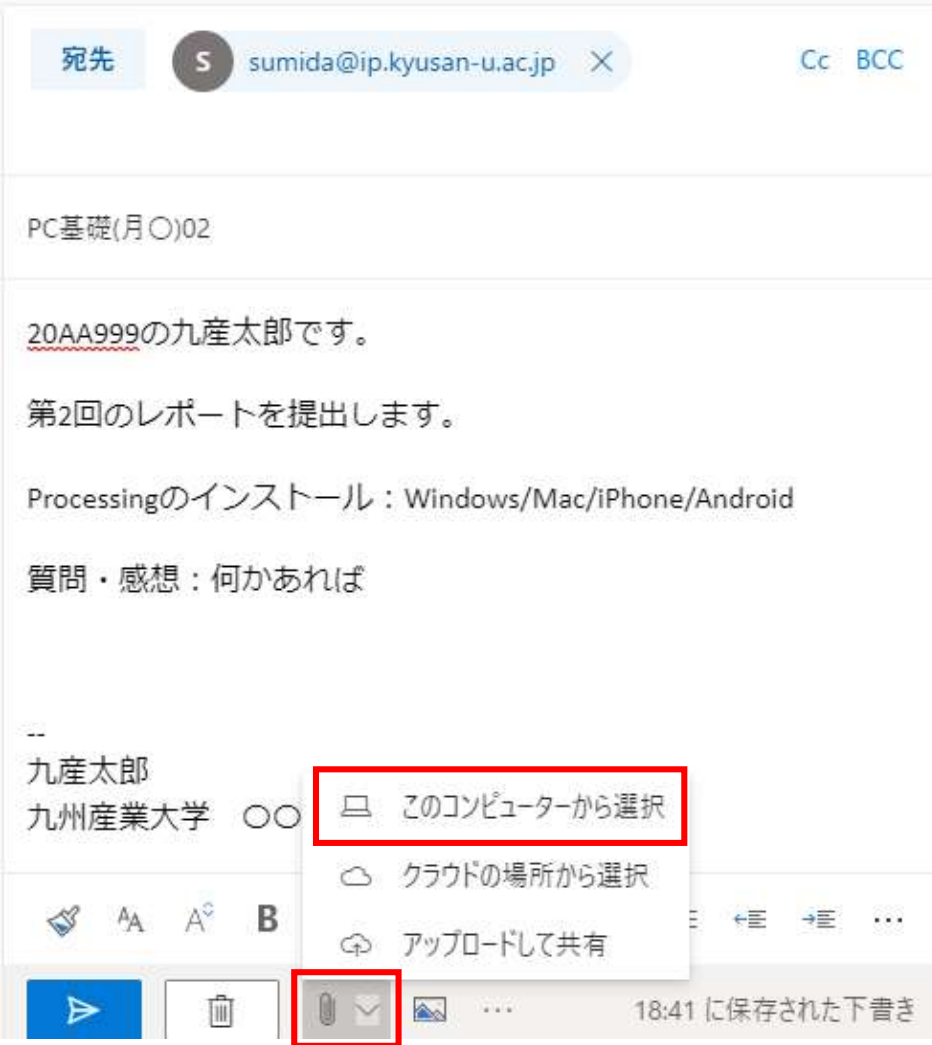
- 画像ファイルを右クリック → 「名前を変更」
 - 「学籍番号.png」に変更



他にも、「スクリーンショット」のアプリを使うのもお勧め
(Mac スクリーンショット アプリ で検索)

スクリーンショットをメールに添付

- メール作成画面の開き方は前回の資料を参照



保存したファイルを選んで「開く」

宛先

S

sumida@ip.kyusan-u.ac.jp X

Cc BCC

PC基礎(月〇)02



20AA999の九産太郎です。

第2回のレポートを提出します。

Processingのインストール：Windows/Mac/iPhone/Android

質問・感想：何かあれば

--
九産太郎

九州産業大学 〇〇学部〇〇学科〇年

✉ A A° B 添付 U ✎ A ☰ ☷ +☷ +☷ ...



18:41 に保存された下書き

実行画面が写っているのを確認して、▶で送信

送ったら、自動返信メールが来るのを確認しておこう。

(返ってこない場合もあるので、送れている自信があればそれでも良い)

レポート

Processingのインストールと実行
スクリーンショットの撮影

iPhone編

Processing iCompilerをインストール

- Processing iCompiler
 - <https://apps.apple.com/jp/app/processing-icompile/id648955851>



Processing iCompiler 4+

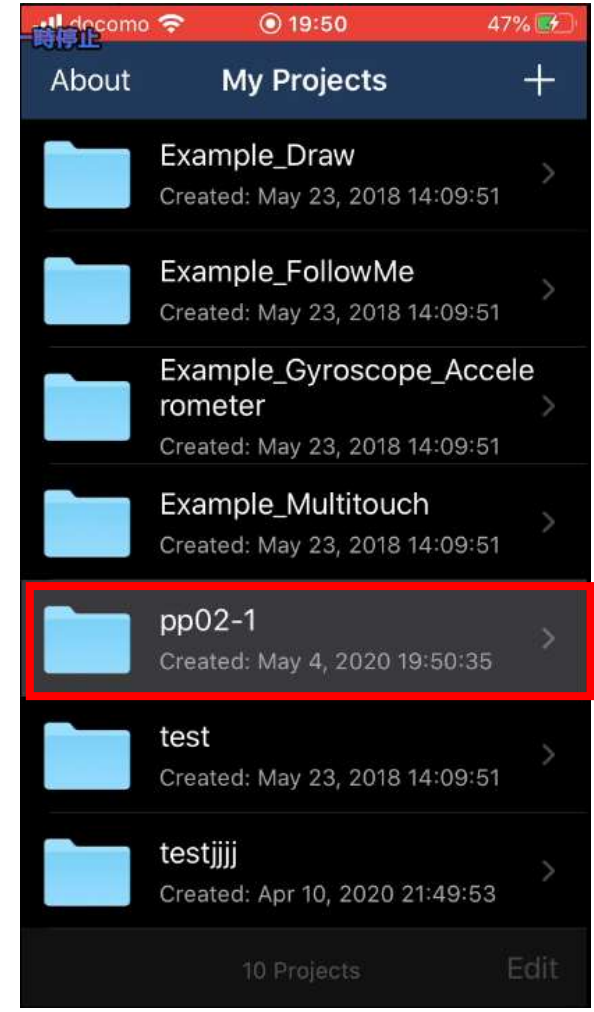
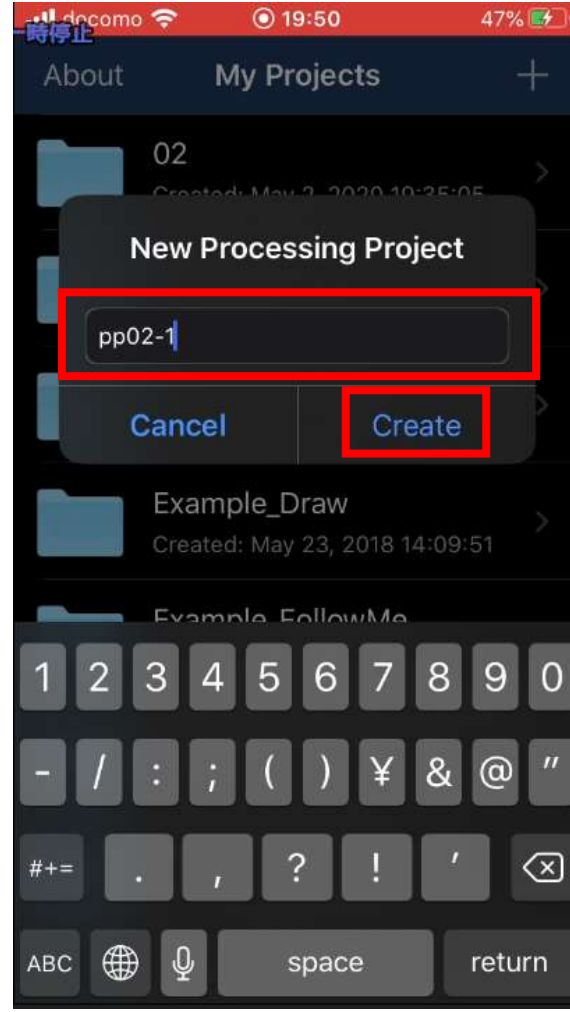
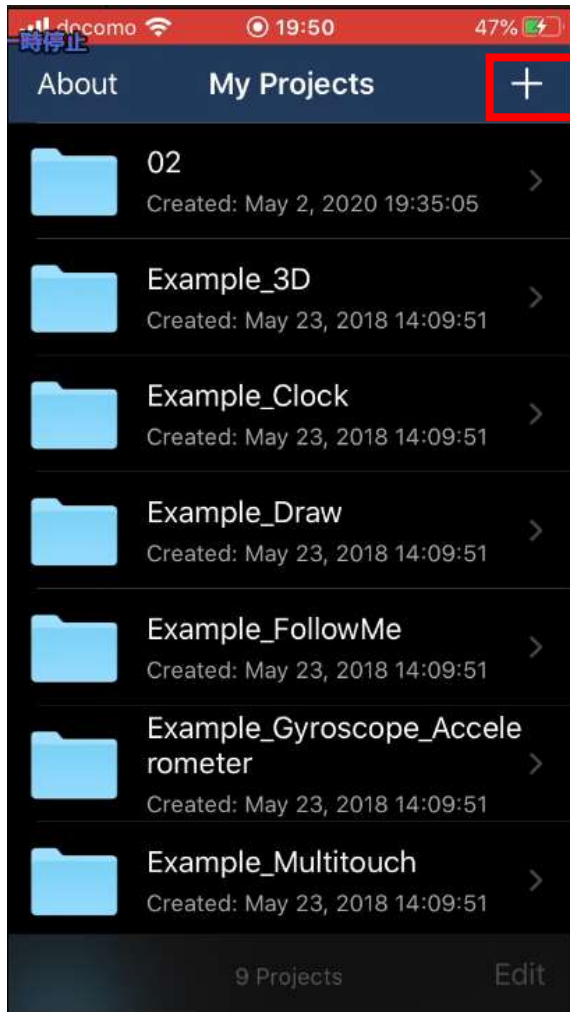
Frogg GmbH

「開発ツール」内2位

★★★★★ 4.0、25件の評価

無料

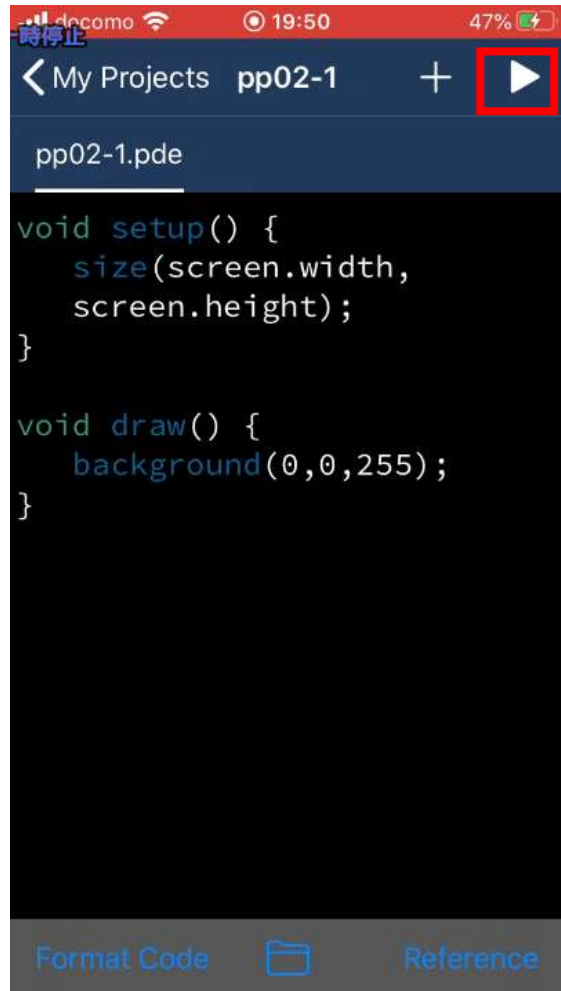
Processing iCompiler 起動とスケッチ作成



「+」タップ → 名前を付ける

作られたフォルダをタップ

Processingを実行



左上のスケッチ名から元に戻る

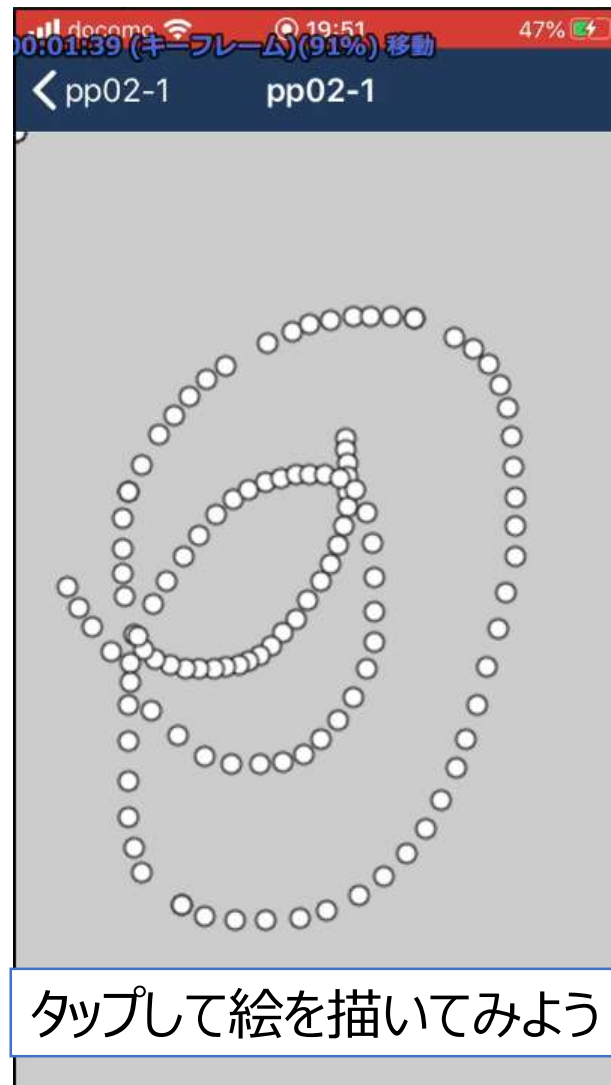


少しプログラムを修正



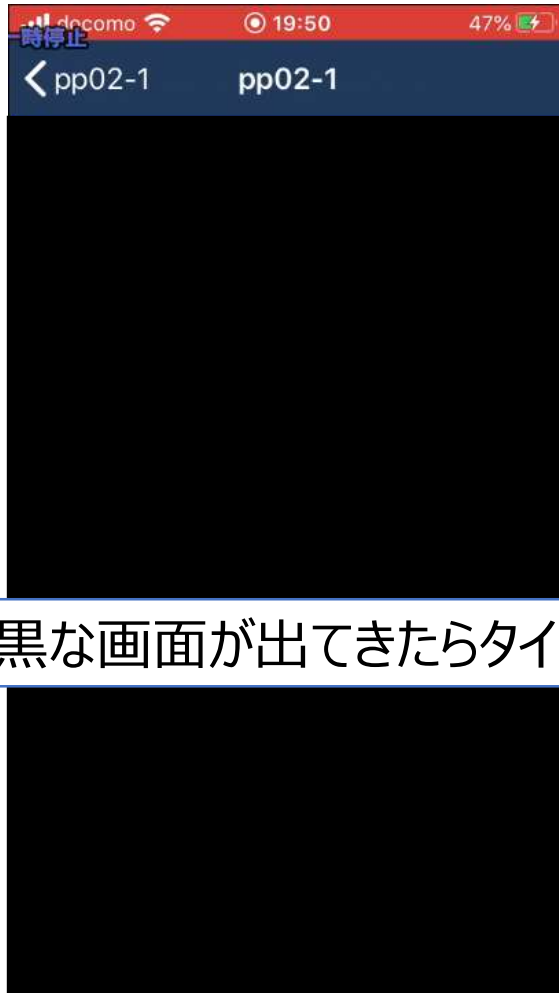
```
void setup() {  
  size(screen.width,  
        screen.height);  
}  
  
void draw() {  
  
  ellipse(mouseX, mouseY, 10,  
          10);  
}
```

この通りにプログラムを修正して、実行



タップして絵を描いてみよう

プログラムが間違っていた時



真っ黒な画面が出てきたらタイプミスがある

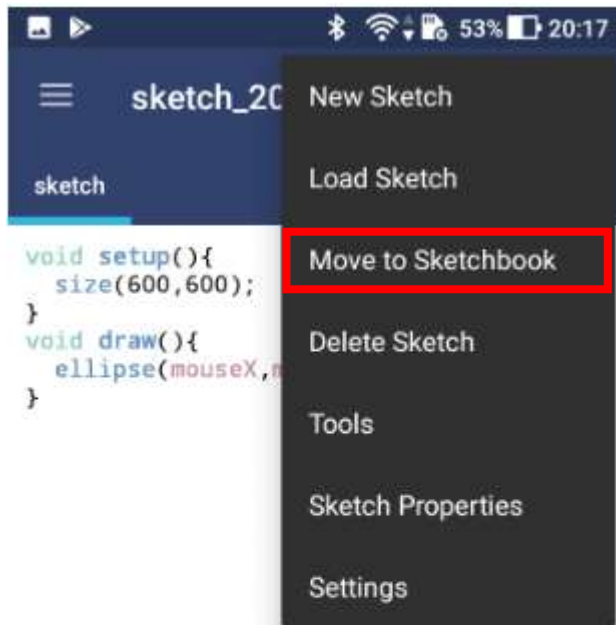


もう一度見直して、再度実行しよう

スクリーンショットを撮影

- 公式のヘルプを参照
 - <https://support.apple.com/ja-jp/HT200289>
- Face ID 搭載モデルの iPhone
 - 音量を上げるボタンとサイドボタンを同時押し
- ホームボタンがあるタイプのiPhone
 - 電源ボタンとホームボタンを同時押し
- 「写真」の中に画像が保存されている

スケッチ（プログラム）を保存



The sketch has been saved.

スクリーンショットをメールに添付

- メール作成画面の開き方は前回の資料を参照



保存したファイルを選ぶ



ファイル添付を確認して
送信

レポート

Processingのインストールと実行
スクリーンショットの撮影

Android編

Androidへのインストール時の注意

- 「提供元不明のアプリ」をインストールする必要がある
- 提供元不明のアプリのインストールを可にしておくのは、セキュリティの面からみて危険なので、必ずインストール後に「提供元不明のアプリ」のインストールを許可しない設定に戻すこと
- どうしても不安なら、インストールせずにOpenProcessingでの実行にしても良い
 - ただし、OpenProcessingでプログラミングをする場合、動作が遅い、安定しない、プログラミングしにくい、などの問題があることは予め理解しておくこと

提供元不明のアプリのインストールを許可

- ・必ず後で「許可しない」に設定を戻すこと



端末によって、設定の場所が異なる場合がある、その場合は各自で調べること

アプリインストール： APDE - Android Processing IDE

- APDE - Android Processing IDE
 - <https://play.google.com/store/apps/details?id=com.calsignlabs.apde&hl=ja>



APDE - Android Processing IDE

CalsignLabs ツール

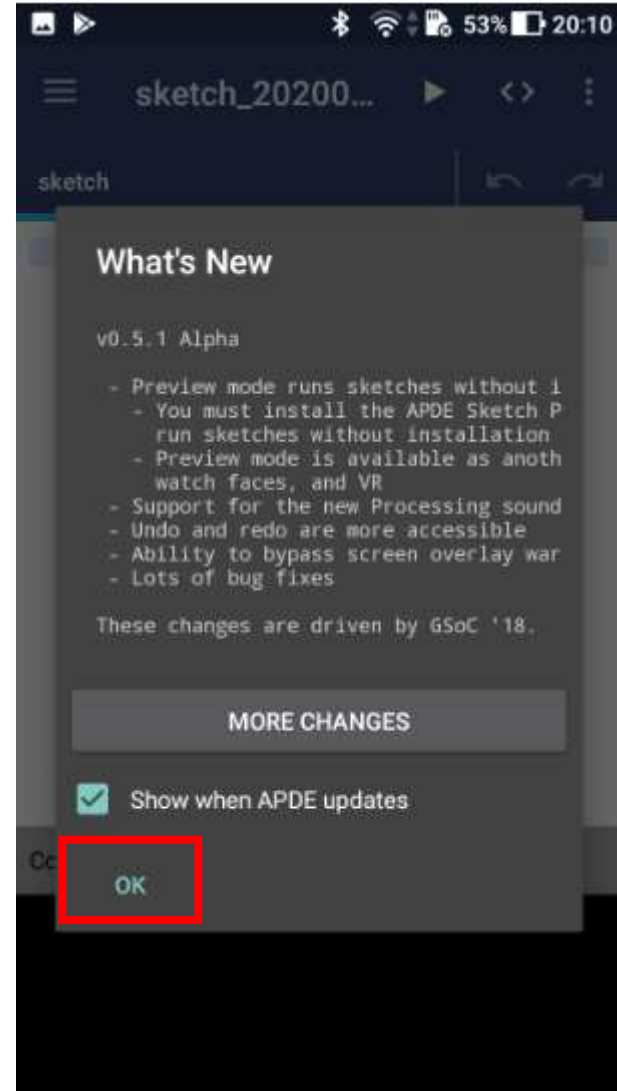
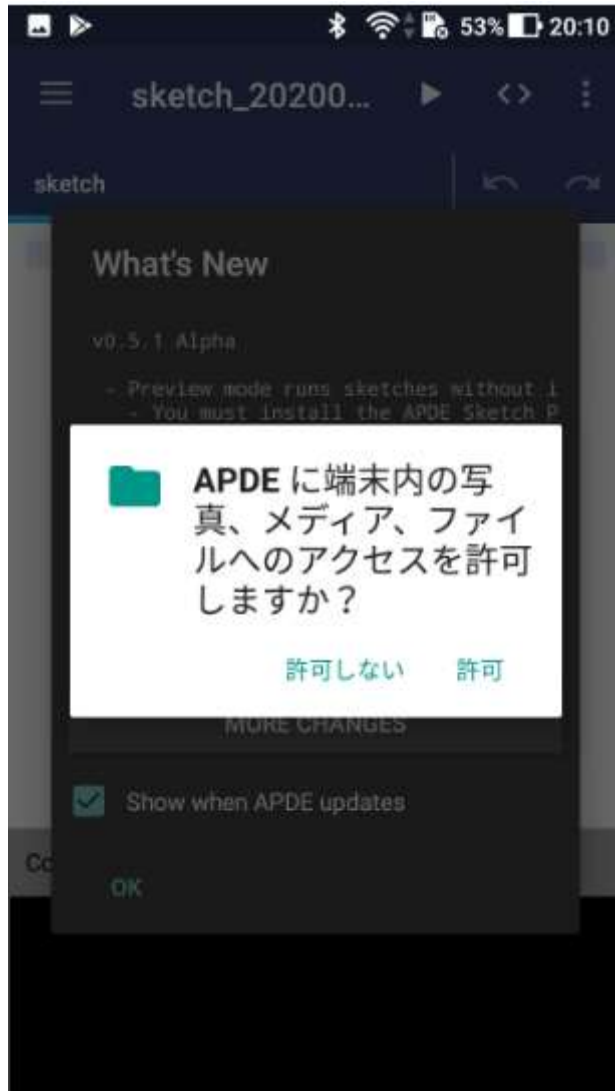
★★★★★ 1,776

3+

このアプリはお使いのすべてのデバイスに対応しています。

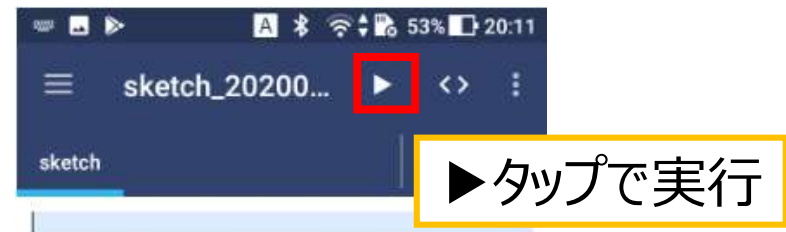
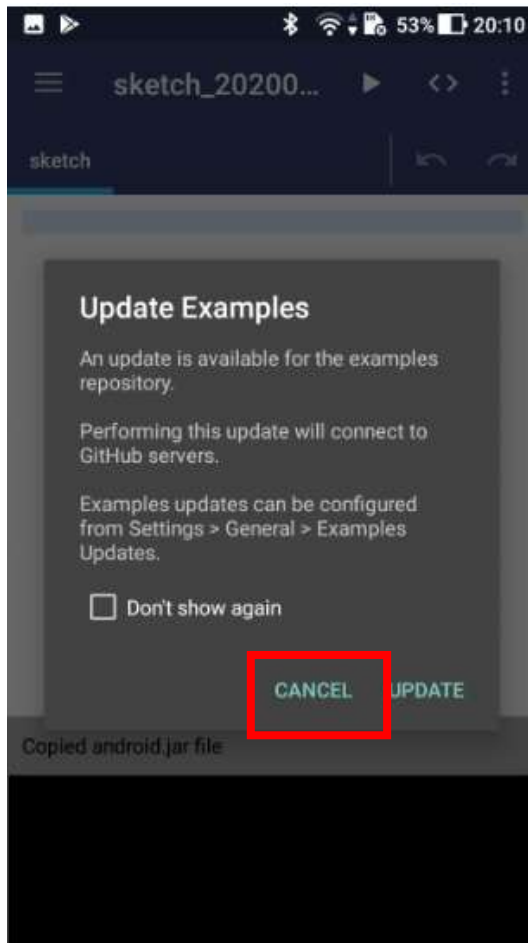
インストール済み

APDEを実行：権限を許可

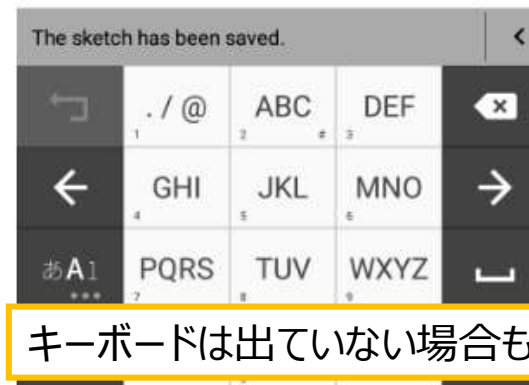


サンプルのアップデート画面

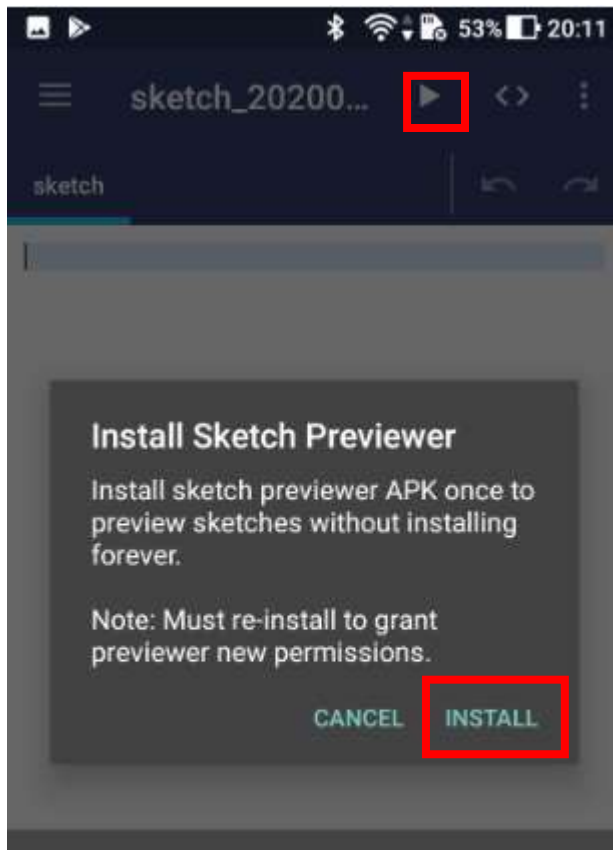
- してもしなくても良い（アップデートする場合の説明は省略）



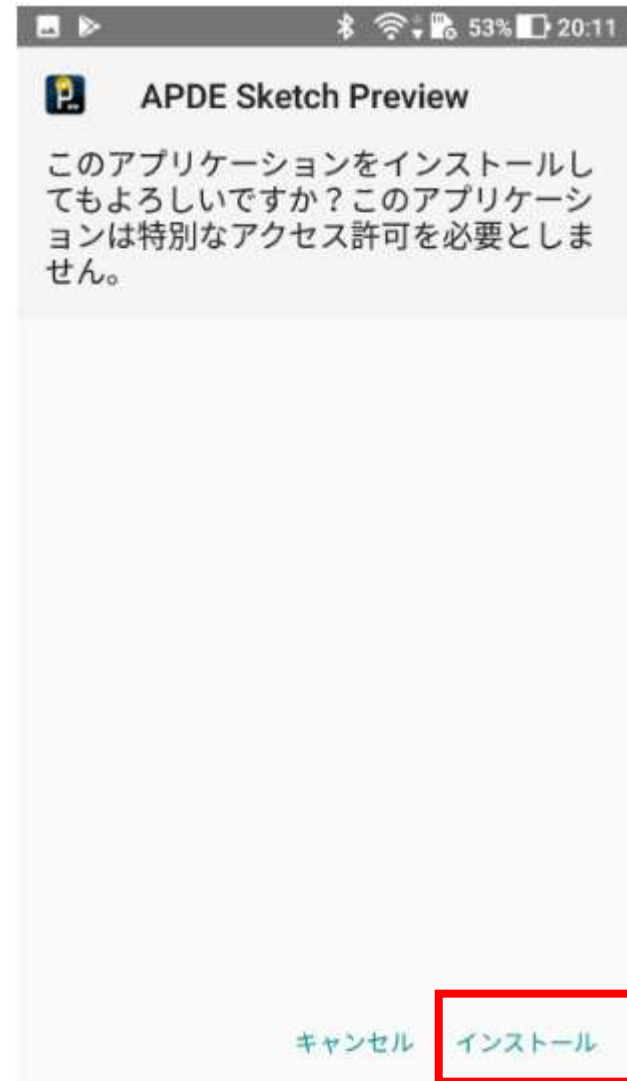
起動直後の画面



APDE Sketch Previewerのインストール



書いたプログラムを動かすためのアプリをインストールする



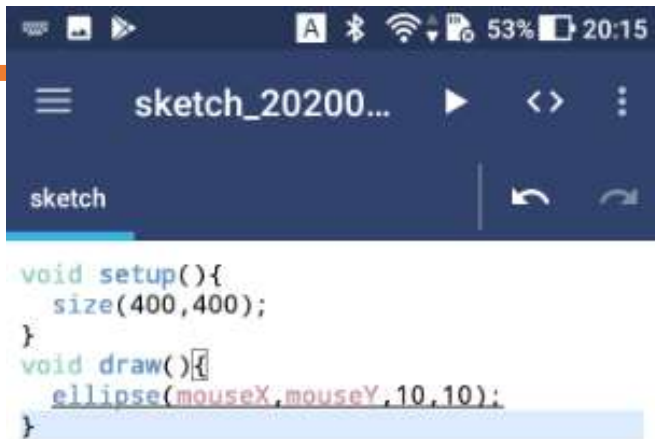
インストールを許可



インストールに成功すると、
薄い灰色(白に見えるかも)
一色の画面が表示される

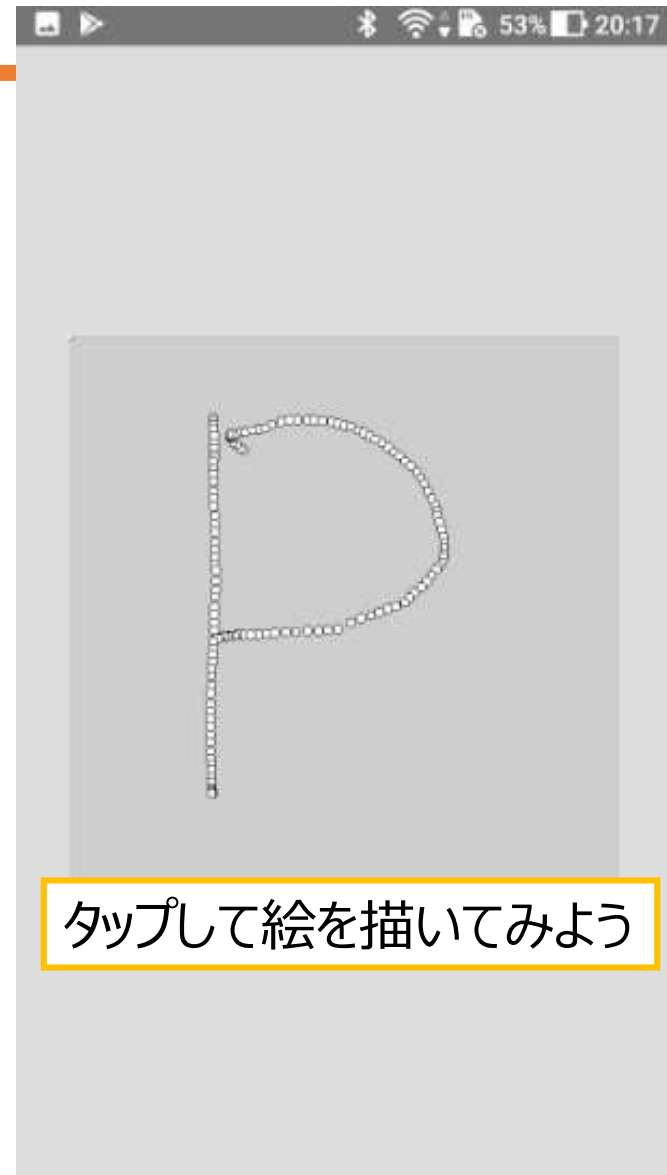
表示を確認したら「戻る」ボタンで戻る

プログラムの入力



```
void setup(){
  size(400,400);
}
void draw(){
  ellipse(mouseX,mouseY,10,10);
}
```

この通りに入力する



提供元不明のアプリのインストールを不許可 (必ず戻しておくように！！)



端末によって、設定の場所が異なる場合がある、その場合は各自で調べること

スクリーンショットを撮影

※機種によって違いがあるので、
上手いいかない場合は各自で調べること

- 公式のヘルプを参照

- <https://support.google.com/android/answer/9075928?hl=ja>

- 電源ボタンと音量小ボタンを同時に数秒ほど長押しします。
- 撮影できない場合は、電源ボタンを数秒ほど長押ししてから、[スクリーンショット] をタップします。
- 上記のいずれの操作でも撮影できない場合は、[スマートフォン
のメーカーのサポートサイト](#)でご確認ください。

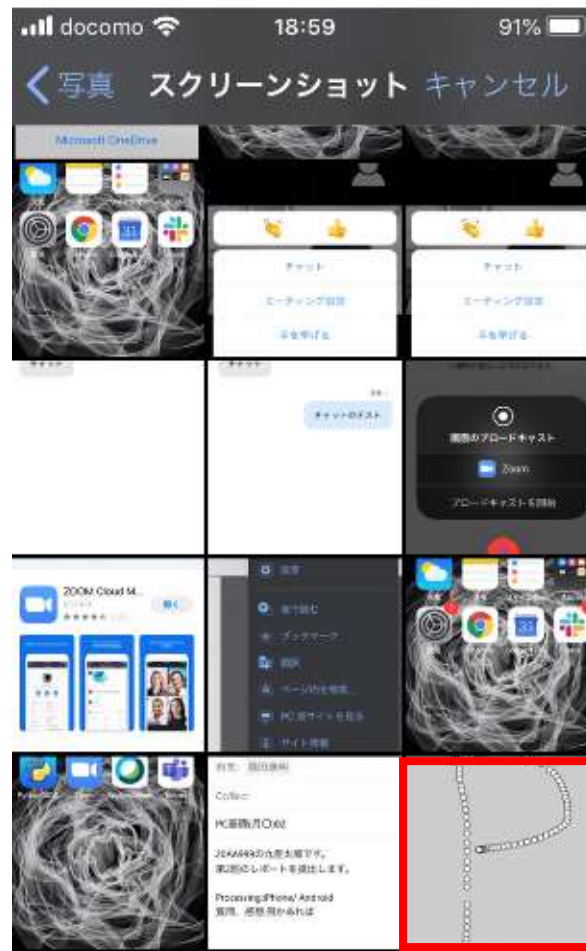
- フォト  アプリに撮影画像が記録されている

スクリーンショットを撮影

- 公式のヘルプを参照
 - <https://support.apple.com/ja-jp/HT200289>
- Face ID 搭載モデルの iPhone
 - 音量を上げるボタンとサイドボタンを同時押し
- ホームボタンがあるタイプのiPhone
 - 電源ボタンとホームボタンを同時押し
- 「写真」の中に画像が保存されている

スクリーンショットをメールに添付

- メール作成画面の開き方は前回の資料を参照



保存したファイルを選ぶ



Processingの実行

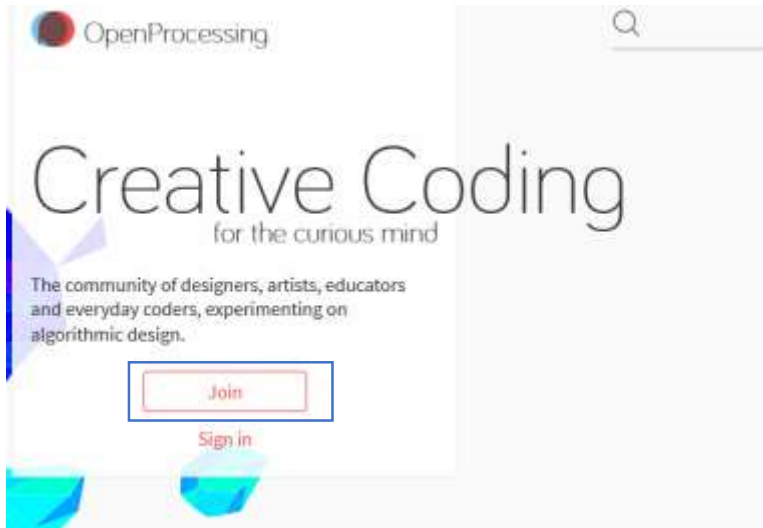
OpenProcessing編

- ブラウザのみで実行する方法
- どうしてもインストール出来ない場合に限る

OpenProcessingでのプログラミング

- OpenProcessing:
<https://www.openprocessing.org/>
 - 世界中のProcessingユーザーが作品を投稿しているサイト
 - オンラインエディタもあり、Processingをインストールしていなくても、Processingでのプログラミングを行える
 - ただし、可能な限りProcessingをインストールしておくこと
 - 通信環境によって動作が安定しないことも多いので、演習では、どうしてもProcessingをインストールできない場合に限り利用すること

OpenProcessingにJoin



Join the Creative Coders
on OpenProcessing

NAME 名前

EMAIL メールアドレス

PASSWORD パスワード

それぞれ入力

STAY SIGNED IN ✕

By clicking Join, you agree to the [Terms of Service](#).

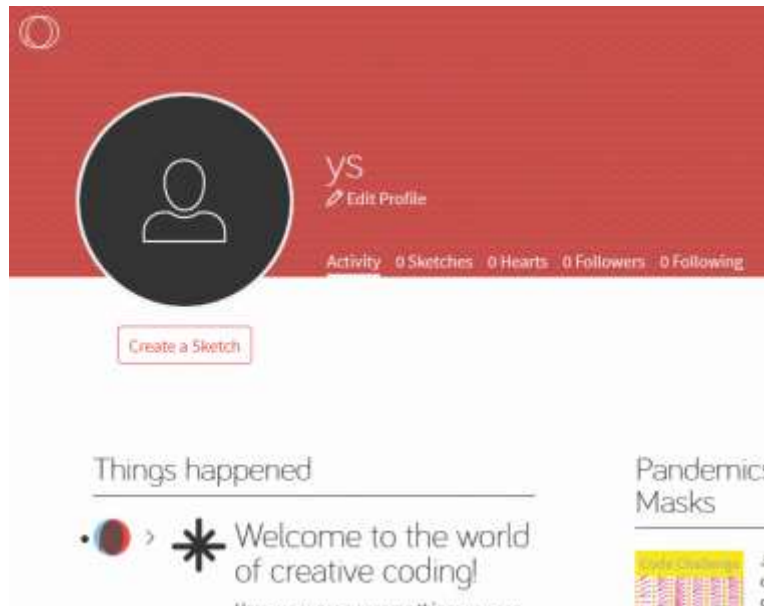
Join

Joinをクリック

Have an account already? [Sign in](#).

一度Joinしたら、次からは Sign in

Join出来たら「Create a Sletch」



スマートフォンの場合はPC版表示

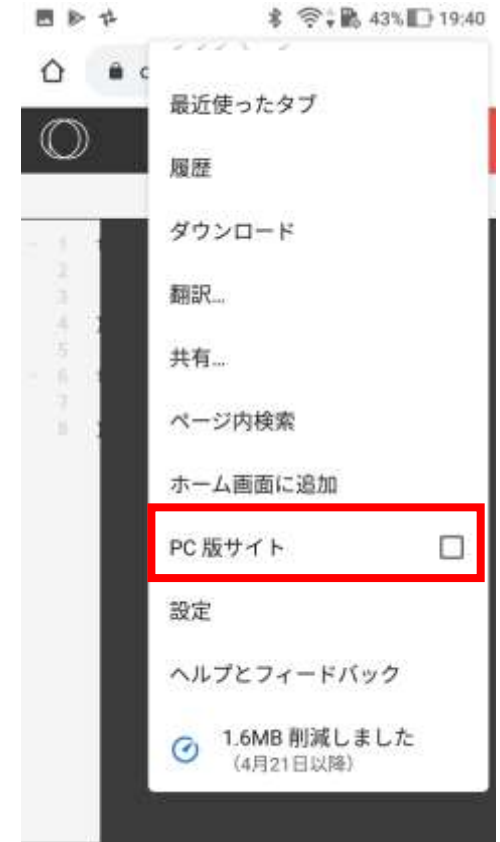
- PC版表示にしないと、スマートフォンでは画面の一部しか表示されない



iPhone版Chromeの場合

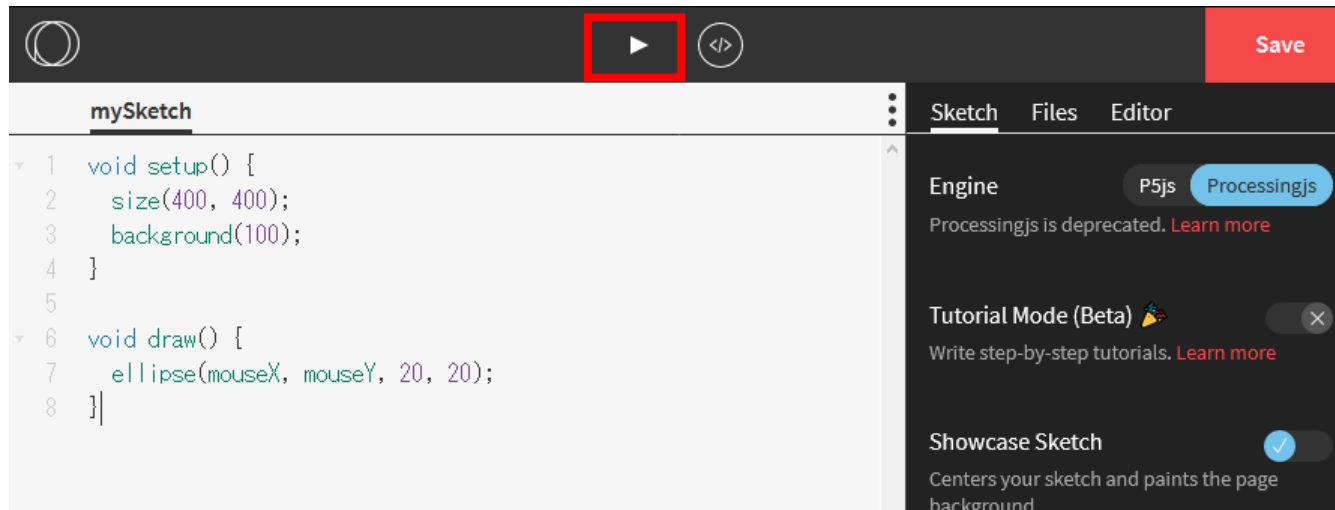


Android版Chromeの場合

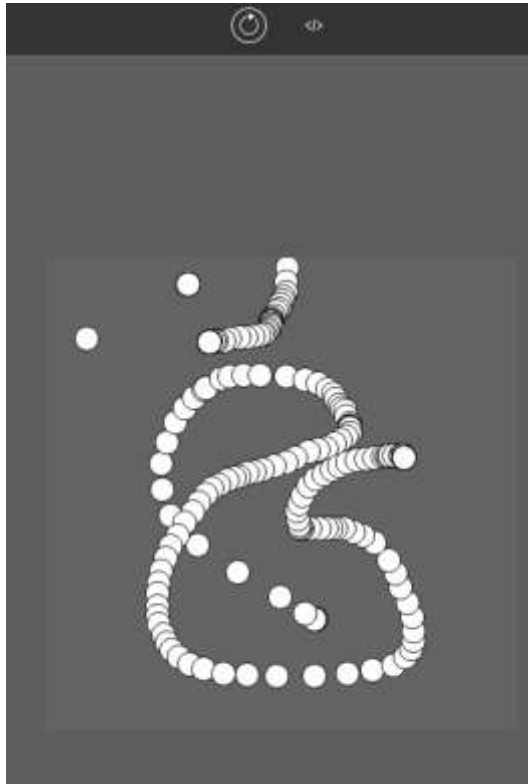


Processingの実行

- EngineをProcessingに変更
- 下記のようにプログラムを修正して、実行 (▶)



マウスを動かして絵を描ければ成功



スクリーンショット撮影、メール添付

- それぞれのOS版を参照